

ФИЛЬТРАЦИЯ ИМПУЛЬСНОГО ШУМА НА ИЗОБРАЖЕНИИ НА ОСНОВЕ АЛГОРИТМА ВЫЯВЛЕНИЯ СООБЩЕСТВ НА ГРАФАХ

С. В. Белим^а, доктор физ.-мат. наук, профессор

С. Б. Ларионов^а, аспирант

^аОмский государственный университет им. Ф. М. Достоевского, Омск, РФ

Постановка проблемы: выявление и устранение импульсного шума на изображениях является актуальной задачей в теории и практике обработки сигналов. Существующие алгоритмы восстановления зашумленных изображений не обеспечивают достаточной точности либо изменяют незашумленные области изображения. **Цель:** разработка алгоритма фильтрации шума с использованием графов. **Результаты:** на базе предложенного алгоритма был реализован метод восстановления изображения, искаженного импульсным шумом. В процессе работы изображение представлялось в виде взвешенного, неориентированного графа, каждая вершина которого соответствует одной точке исходного изображения. К построенному графу применялся алгоритм выделения сообществ. Поврежденными считаются точки, которые соответствуют сообществам, состоящим из одной вершины. Реализованный метод был протестирован с помощью компьютерного эксперимента, проведенного как на черно-белых и на цветных искусственных изображениях, так и на фотографиях. Выявлено, что предложенный метод на 20 % эффективнее медианного фильтра при любом уровне зашумления. Более высокая эффективность предложенного метода обусловлена тем, что сначала ищутся поврежденные точки и потом восстанавливаются только они, тогда как большинство фильтров в процессе работы изменяют все точки изображения. Показана зависимость эффективности предложенного метода от процента шума на изображении.

Ключевые слова — импульсный шум, фильтрация изображений, выделение сообществ на графах.

Введение

Импульсный шум на цифровых изображениях проявляется как случайное изменение цвета отдельных, случайно расположенных пикселей, которые принято называть испорченными пикселями [1, 2]. Наличие шумов на изображении не только влияет на его визуальное восприятие, но и сказывается на результатах работы алгоритмов сегментации, выделения контуров, распознавания образов и т. д.

Для устранения влияния шумов широкое распространение получили сглаживающие фильтры. Наиболее часто используются фильтр Винера и медианный фильтр [3]. Следует также выделить нелокальные методы фильтрации [4–6], демонстрирующие более качественное подавление импульсного шума. Тем не менее сглаживающие фильтры обладают существенным недостатком, состоящим в изменении всего изображения в целом, а не только поврежденных пикселей. Данные изменения приводят к размытию контуров областей изображения и затрудняют его дальнейшую обработку.

Для минимизации влияния фильтра на не испорченную часть изображения используется подход, использующий поиск пикселей, измененных импульсным шумом. В дальнейшем действие фильтра ограничивается только испорченными пикселями. Задача поиска поврежденных пикселей сама по себе является достаточно сложной и решается, как правило, опираясь на анализ изображения методами Data Mining. В таком под-

ходе алгоритм устранения импульсного шума состоит из двух этапов. На первом этапе выполняется поиск поврежденных пикселей. На втором этапе выбирается новый цвет для каждого поврежденного пикселя. Для поиска поврежденных пикселей осуществляется последовательный проход по всему изображению с анализом цвета каждой точки. Принятие решения о поврежденности пикселя импульсным шумом принимается исходя из анализа его ближайших соседей, т. е. пикселей, соприкасающихся с данным пикселем по одной стороне либо по вершине.

Алгоритмы выявления испорченных пикселей можно разделить на две группы. Первая группа алгоритмов предназначена для выявления импульсного шума типа «Соль и перец». Особенностью данного вида шума является то, что поврежденные пиксели имеют либо максимальное значение цвета в палитре, либо минимальное. Соответственно, алгоритмы выявления поврежденных пикселей базируются на этой априорной информации [7–9]. Однако даже эти алгоритмы не гарантируют стопроцентное обнаружение поврежденных пикселей. Вторая группа алгоритмов ориентирована на импульсный шум случайного значения. Для импульсного шума случайного значения наибольшее распространение получили алгоритмы, использующие схему SD-ROM [10, 11]. Основная идея схемы SD-ROM заключается в анализе пикселей внутри скользящего окна размером 3×3 и принятия решения о поврежденности пикселя в центре этого окна. Алгоритм принятия решения в SD-ROM представ-

ляет собой пороговую схему. В работе [12] предложена схема принятия решения, включающая в себя метод анализа иерархий, при этом также используется скользящее окно. Повысить эффективность поиска поврежденных пикселей удастся путем перехода от анализа небольшой окрестности каждой точки к анализу изображения в целом. В статье [13] для поиска поврежденных пикселей используется метод построения ассоциативных правил. Для выявления правил с высоким уровнем поддержки анализируется все изображение. В работе [14] для выявления поврежденных пикселей используется алгоритм сегментации изображения. Импульсный шум детектируется на основе поиска сегментов, включающих в себя один пиксель.

Алгоритмы устранения импульсного шума на изображении с известным списком испорченных пикселей сводятся к заполнению таблицы с пропусками. Наиболее простой подход состоит в выборе цвета по среднему значению ближайших соседей. Такой метод дает неплохие результаты на областях равномерной заливки и совершенно не приемлем при наличии резких переходов цвета, так как приводит к размытию границ. Более действенен подход, основанный на линейных многообразиях [15]. Также для восстановления пропущенных данных могут быть использованы искусственные нейронные сети [16]. Эти алгоритмы основаны, как правило, на данных из локальной окрестности поврежденного пикселя. Защита изображений, передаваемых по каналам связи, может также осуществляться с помощью специфических преобразований, например стрип-метода [17–19]. Для устранения импульсного шума могут использоваться те же алгоритмы, что и для поиска поврежденных пикселей. В статье [20] представлен алгоритм восстановления, основанный на ассоциативных правилах, в работе [21] — алгоритм устранения импульсного шума на основе метода анализа иерархий.

Как показывает изучение перечисленных работ, методы, основанные на анализе всего изображения, обеспечивают более качественное устранение импульсного шума по сравнению с локальными методами. Целью данной статьи является разработка и апробация алгоритма устранения пикселей, поврежденных импульсным шумом, на основе метода выделения сообщества на графах, хорошо зарекомендовавшего себя в задаче сегментации изображения [22].

Постановка задачи и алгоритм фильтрации

Пусть на входе алгоритма задано изображение с импульсным шумом размерами $N \times M$ пикселей. Для определения положения пикселя на изображении будем использовать пару целых чисел (x, y) ,

представляющих собой его координаты. Число x принимает целые значения на отрезке $[0, N - 1]$, число y — целые значения на отрезке $[0, M - 1]$. В общем случае цветного изображения пиксель с координатами (x, y) характеризуется тремя цветовыми составляющими: $r(x, y)$ — интенсивность красного цвета, $g(x, y)$ — интенсивность зеленого цвета, $b(x, y)$ — интенсивность синего цвета.

Сопоставим изображению неориентированный взвешенный граф G . Каждому пикселю изображения будет соответствовать вершина графа G . Соединим ребрами только вершины, соответствующие пикселям, являющимся ближайшими соседями на изображении. Ближайшими соседями считаются пиксели, одна или обе координаты которых отличаются не более чем на единицу. Вес ребер будем вычислять на основе цветовых компонент пикселей, соответствующих концам ребра. Для двух соседних вершин $v_i = (x_i, y_i)$ и $v_j = (x_j, y_j)$ вес ребра будет равен

$$d(v_i, v_j) = \exp\left(-\frac{1}{h} \sqrt{(r_i - r_j)^2 + (g_i - g_j)^2 + (b_i - b_j)^2}\right).$$

Здесь $r_i = g(v_i, v_j)$, $g_i = g(v_i, v_j)$, $b_i = b(v_i, v_j)$. Настраиваемый параметр h определяет значение разности цвета между соседними пикселями, соответствующее переходу через границу двух сегментов. Данный параметр определяется пользователем и является общим для всего изображения. Как показано в работах [14, 18], такой вид весовой функции позволяет достаточно точно различать изменение цвета, соответствующее границам областей на изображении. Будем разбивать граф на подграфы, вершины которых связаны между собой сильнее, чем с остальными. Такие подграфы принято называть сообществами (community). Качество разбиения графа количественно может быть оценено с помощью функции модульности Ньюмана [23, 24]. Чем больше значение функции модульности, тем более качественно осуществлено разбиение. Поврежденными будем считать пиксели, объединение которых с любыми другими сообществами понижает значение функции модульности. Таким образом, будут выделены пиксели, образующие сообщества из одной вершины, т. е. сильно отличающиеся от своих ближайших соседей. Опишем эту процедуру формально.

Определим матрицу весов E для графа G . Значения диагональных элементов E_{ii} равны весам вершин. В начале работы алгоритма вес всех вершин нулевой. Остальные элементы матрицы весов E_{ij} ($i \neq j$) равны весу соответствующего ребра. Следует отметить, что в матрице E_{ij} будет много нулевых элементов, так как в графе G ребрами соединены только вершины, соответствующие ближайшим соседям на изображении. Матрица E будет симметричной относительно

главной диагонали, так как граф G неориентированный. Перейдем к приведенному виду матрицы весов $e = E/m$, где $m = \sum_{i,j=1}^{MN} E_{ij}$. Элемент e_{ij} равен доле веса заданного ребра в общем весе графа. В дальнейшем под матрицей весов будем понимать именно приведенный вид. Легко убедиться, что $\sum_{i,j=1}^{MN} e_{ij} = 1$.

Величина модульности определяется выражением [24, 25]

$$Q(G) = \sum_{i=1}^K e_{ij} - \sum_{i=1}^K a_i b_i,$$

где K — количество вершин графа; a_i — приведенная исходящая степень вершины v_i ($a_i = \sum_{j=1, j \neq i}^K e_{ij}$); b_i — приведенная входящая степень вершины v_i ($a_i = \sum_{j=1, j \neq i}^K e_{ji}$). Рассматриваемый граф является неориентированным, поэтому входящая и исходящая степени для всех вершин одинаковы ($a_i = b_i, i = 1, \dots, K$). Функция модульности принимает более простой вид

$$Q(G) = \sum_{i=1}^K e_{ij} - \sum_{i=1}^K a_i^2.$$

Для поиска сообществ на графе используем процедуру образования стяжек. Под стяжкой понимается преобразование, при котором некоторый подграф H графа G заменяется одной вершиной v_H . Если какая-то из вершин подграфа H была связана ребром с вершиной v из подграфа $G \setminus H$, то вершина v_H также будет связана ребром того же веса с вершиной v . Вес новой вершины будет равен сумме весов ребер и вершин, входящих в подграф H . Новый граф обозначим G_H . Подграф H будем считать сообществом, если $Q(G_H) > Q(G)$. Следует отметить, что при образовании стяжки уменьшается число вершин графа K . Нашей задачей является поиск вершин графа, не входящих ни в одно более крупное сообщество, проводимый с помощью следующего алгоритма.

1. Осуществляем последовательный проход по всем пикселям изображения.

2. Для каждого пикселя v поочередно рассматриваем ближайшие соседние пиксели $v^{(i)}$ ($i = 1, \dots, 8$). Рассматриваем подграфы, состоящие из двух вершин v и $v^{(i)}$ ($i = 1, \dots, 8$) и пытаемся объединить их в сообщество. При каждом объединении вычисляем изменение функции модульности ΔQ_i ($i = 1, \dots, 8$).

3. Если при объединении данной вершины в сообщества со всеми ближайшими соседями изменение функции модульности отрицательно ($\Delta Q_i < 0, i = 1, \dots, 8$), то соответствующий пиксель считаем поврежденным.

Изменение функции модульности достаточно быстро вычисляется из текущих характеристик графа, соответствующего изображению. При объединении вершин v_i и v_j изменение модульности будет иметь вид

$$\Delta Q = 2(e_{ij} - a_i a_j).$$

Очевидно, что предложенный алгоритм имеет линейную трудоемкость от числа пикселей.

После выявления поврежденных пикселей необходимо выбрать для них цвет, исследовав окружающие пиксели. Для этого оценим ближайших соседей. Пусть минимальное значение цветовой компоненты пикселей, соседних с поврежденным пикселем, равно m_1 , а максимальное — m_2 . Проведем последовательный перебор всех значений цвета поврежденного пикселя от m_1 до m_2 . Для каждого значения будем вычислять значение изменения функции модульности ΔQ при объединении данной вершины в сообщество с одним из ближайших соседей. В качестве окончательного цвета оставим тот, который позволяет получить максимальное значение изменения функции модульности.

Компьютерный эксперимент

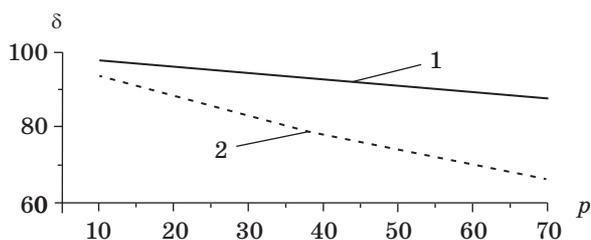
Компьютерный эксперимент проводился как на искусственных изображениях геометрических объектов, так и на цветных фотографиях. Уровень импульсного шума характеризовался величиной p , показывающей процент поврежденных пикселей по отношению к общему количеству пикселей изображения. Импульсный шум моделировался линейным конгруэнтным генератором псевдослучайных чисел, с помощью которого определялись как координаты поврежденных пикселей, так и их цвет. При проведении эксперимента количество испорченных пикселей варьировалось от 10 до 70 %. Улучшение изображения производилось посредством предложенного фильтра, а также, для сравнения, широко распространенного медианного фильтра.

Для сравнения близости изображений использовалась метрика Минковского [25, 26], согласно которой расстояние между изображениями A и C находится по формуле

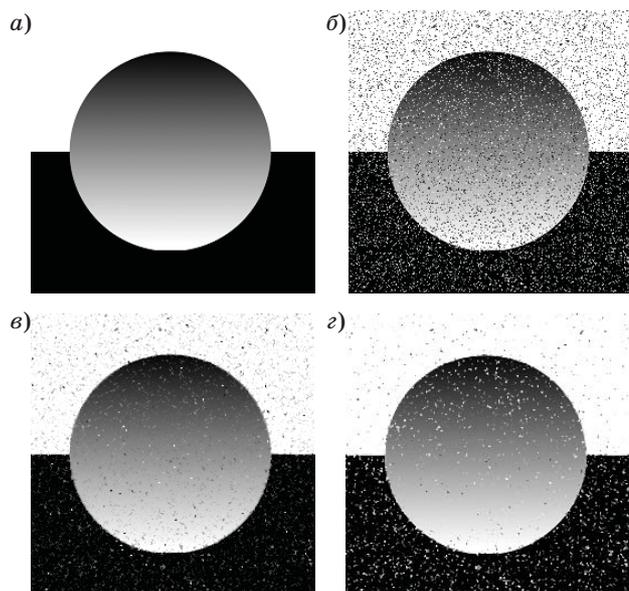
$$d(A, C) = \max_{n,m} \sum_{k=1}^N \frac{1}{N} |A_{nm}^{(k)} - C_{nm}^{(k)}|,$$

где A_{mn} и C_{mn} — значения цветов пикселей изображения A и C ; N — количество пикселей.

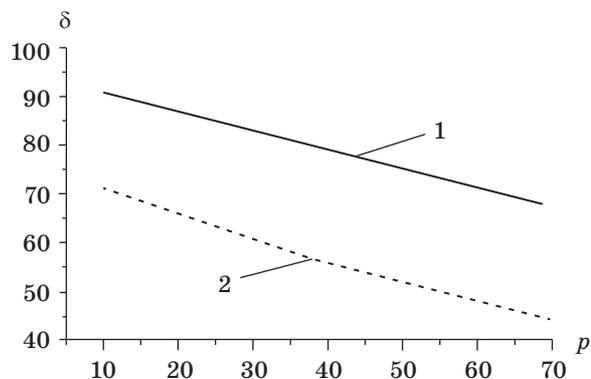
Относительное улучшение изображения вычислялось на основе расстояния $d(orig_fig, r_fig)$ от восстановленного изображения r_fig до исходного $orig_fig$ и расстояния $d(orig_fig, p_fig)$ от ис-



■ **Рис. 1.** Зависимость относительного улучшения изображения от процента зашумления для предложенного (1) и медианного (2) фильтра



■ **Рис. 2.** Результаты применения фильтра к искусственному изображению с уровнем шума $p = 20\%$: *a* — исходное изображение; *b* — зашумленное изображение; *v* — изображение, восстановленное предложенным фильтром; *z* — изображение, восстановленное медианным фильтром



■ **Рис. 3.** Зависимость относительного улучшения от процента зашумления искусственного изображения для предложенного (1) и медианного (2) фильтра



■ **Рис. 4.** Результаты применения фильтра к изображению «Lena» с уровнем шума $p = 20\%$: *a* — исходное изображение; *b* — зашумленное изображение; *v* — изображение, восстановленное предложенным фильтром; *z* — изображение, восстановленное медианным фильтром

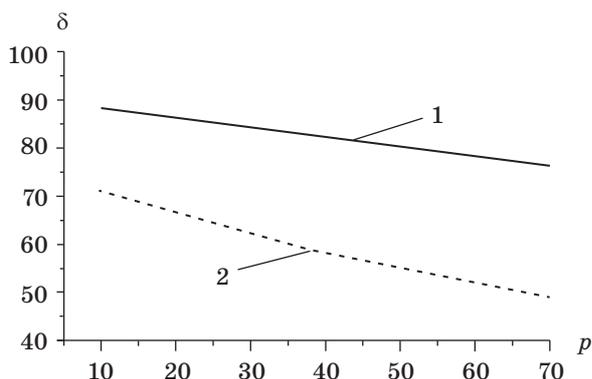
порченного изображения p_fig до исходного изображения $orig_fig$:

$$\delta = \frac{d(orig_fig, p_fig) - d(orig_fig, r_fig)}{d(orig_fig, p_fig)} \cdot 100\%.$$

Эксперимент для прямоугольной области с равномерной заливкой показал, что предложенный фильтр позволяет значительно улучшить изображение. Зависимость относительного улучшения от процента испорченных пикселей для предложенного и медианного фильтров представлена на рис. 1.

Результаты применения предложенного и медианного фильтров для улучшения искусственного изображения с наличием сплошной и градиентной заливки показаны на рис. 2, *a–z*.

Хорошо видно, что результаты предложенного фильтра более выигрышно выглядят для темных участков изображения, тогда как медианный фильтр дает лучший визуальный результат для светлой части изображения. Этот эффект связан с выбором нового цвета испорченного пикселя. При использовании медианного фильтра цвета восстановленных пикселей смещены в область белого цвета. При этом цвета окружающих его пикселей также изменяют свой цвет. Численное сравнение результатов работы показывает значительное преимущество предложенного алгоритма перед медианым фильтром.



■ Рис. 5. Зависимость относительного улучшения от процента зашумления изображения «Lena» с помощью предложенного (1) и медианного (2) фильтра

Зависимость относительного улучшения от процента зашумления искусственного изображения представлена на рис. 3.

Также данный фильтр позволяет получить значительно лучшие результаты и для фотографических изображений. Результаты для хорошо известного изображения «Lena» показаны на рис. 4, а–г.

Хорошо известно, что изображение «Lena» характеризуется большим количеством мелких деталей, которые создают сложности для всех фильтров. Предложенный фильтр дает значительно лучшие результаты даже при визуальном сравне-

нии. Зависимость относительного улучшения от процента зашумления представлена на рис. 5.

Заключение

Таким образом, предложенный фильтр обладает хорошими характеристиками при линейной трудоемкости. Как видно из графиков (см. рис. 1, 3 и 5), эффективность данного фильтра примерно на 20 % выше, чем медианного при любом проценте испорченных пикселей. Такое заметное преимущество объясняется тем, что обычные фильтры изменяют все пиксели изображения. Исправление испорченных пикселей приближает изображение к оригиналу, но при этом изменение неиспорченных пикселей увеличивает расстояние до оригинала. Данное свойство присуще не только медианному фильтру, но и всем традиционным фильтрам.

Предложенный в данной статье фильтр действует избирательно и изменяет только те пиксели, которые значительно отличаются от окружающих. С высокой вероятностью такие пиксели окажутся поврежденными импульсным шумом. Выбор нового цвета на основе присоединения к одному из соседних сообществ пикселей позволяет формировать группы близких по характеристикам точек. Также следует отметить высокое быстродействие фильтра, связанное с линейной трудоемкостью алгоритма, лежащего в его основе. Время обработки одного изображения в пределах погрешности одинаково с медианным фильтром.

Литература

1. Pitas I., Venetsanopoulos A. Nonlinear Digital Filters: Principles and Applications. — Boston: Kluwer Academic Publishers, 1990. — 391 p.
2. Huang T. S. Two-dimensional Digital Signal Processing II: Transforms and Median Filters. — Berlin a.o: Springer, 1981. — 391 p.
3. Boyle R., Sonka M., Hlavac V. Image Processing, Analysis, and Machine Vision, First Edition. — Cambridge: University Press, 2008. — 920 p.
4. Mitra S., Sicuranza G. Nonlinear Image Processing. — Academic Press, 2001. — 455 p.
5. Buades A., Morel J. A Non-Local Algorithm for Image Denoising // IEEE Computer Society Conf. on Computer Vision and Pattern Recognition 2, Washington DC, June 20–26, 2005. P. 60–65.
6. Dabov K., Foi A., Katkovnik V., Egiazarian K. Image Denoising by Sparse 3-D Transform-domain Collaborative Filtering // IEEE Transactions on Image Processing. 2007. N 8(16). P. 2080–2095. doi:10.1109/TIP.2007.901238
7. Chan R., Ho C., Nikolova M. Salt-and-pepper Noise Removal by Median-type Noise Detectors and De-
- tail-preserving Regularization // IEEE Transactions on Image Processing. 2005. N 10(14). P. 1479–1485. doi:10.1109/TIP.2005.852196
8. Kam H. S., Tan W. H. Noise Detection Fuzzy (NDF) Filter for Removing Salt and Pepper Noise // IVIC 2009: Visual Informatics: Bridging Research and Practice, Kuala Lumpur, November 11–13, 2009. P. 479–486. doi:10.1007/978-3-642-05036-7_45
9. Najeer A. J., Rajamani V. Design of Hybrid Filter for Denoising Images Using Fuzzy Network and Edge Detecting // American Journal of Science Research. 2009. N 3. P. 5–14.
10. Abreu E., Lightstone M., Mitra S. K., Arakawa S. K. A New Efficient Approach for the Removal of Impulse Noise from Highly Corrupted Images // IEEE Transactions on Image Processing. 1996. N 6(5). P. 1012–1025. doi:10.1109/83.503916
11. Garnett R., Huegerich T., Chui C., He W. A Universal Noise Removal Algorithm with an Impulse Detector // IEEE Trans Image Process. 2005. N 14(11). P. 1747–1754. doi:10.1109/TIP.2005.8572.61
12. Белим С. В., Селиверстов С. А. Использование метода анализа иерархий для выявления импульсно-

- го шума в графических объектах // Информационные технологии. 2015. № 4. С. 251–258.
13. **Белим С. В., Майоров-Зильбернагель А. О.** Алгоритм поиска поврежденных пикселей и удаления импульсного шума на изображениях с использованием метода ассоциативных правил // Наука и образование: электронное научно-техническое издание. 2014. № 12. <http://technomag.bmstu.ru/doc/744983.html> (дата обращения: 08.02.2017). doi:10.7463/1214.0744983
 14. **Белим С. В., Кутлуниин П. Е.** Выявление поврежденных пикселей на изображении с помощью алгоритма кластеризации // Вестник компьютерных и информационных технологий. 2016. № 3. С. 3–10. doi:10.14489/vkit.2016.03.pp.003-010
 15. **Ларионов И. Б.** Многомерные линейные многообразия как способ восстановления графической информации // Математические структуры и моделирование. 2010. № 1(21). С. 24–31.
 16. **Ларионов И. Б.** Карты Кохонена как способ восстановления мультимедийной информации // Журнал радиоэлектроники. 2010. № 10. С. 29.
 17. **Mironovsky L. A., Slaev V. A.** Strip-method for Image and Signal Transformation. — Berlin: De Gruyter, 2011. — 160 p.
 18. **Мироновский Л. А., Слаев В. А.** Двустороннее поперечное стрип-преобразование и его корневые изображения // Измерительная техника. 2012. № 10. С. 6–10.
 19. **Мироновский Л. А., Слаев В. А.** Стрип-метод помехоустойчивого преобразования изображений // Измерительная техника. 2006. № 8. С. 6–12.
 20. **Белим С. В., Майоров-Зильбернагель А. О.** Восстановление изображений со статическими пропусками на основе метода ассоциативных правил // Вестник компьютерных и информационных технологий. 2014. № 12. С. 18–23. doi:10.14489/vkit.2014.12.pp.018-023
 21. **Белим С. В., Селиверстов С. А.** Алгоритм восстановления поврежденных пикселей на зашумленных изображениях на основе метода анализа иерархий // Наука и образование: электронное научно-техническое издание. 2014. № 11. <http://technomag.bmstu.ru/doc/742145.html> (дата обращения: 20.01.2017). doi:10.7463/1114.0742145
 22. **Белим С. В., Ларионов С. Б.** Алгоритм сегментации изображений, основанный на поиске сообществ на графах // Компьютерная оптика. 2016. № 6. С. 904–910. doi:10.18287/2412-6179-2016-40-6-904-910
 23. **Newman M. E.** Analysis of Weighted Networks // Physical Review E. 2004. N 5(70). P. 056131. doi:10.1103/PhysRevE.70.056131
 24. **Cluset A., Newman M. E. J., Moore C.** Finding Community Structure in Very Large Networks // Physical Review E. 2004. N 6(70). P. 066111. doi:10.1103/PhysRevE.70.066111
 25. **DiGesù V., Staravoirov V. V.** Distance-based Functions for Image Comparison // Pattern Recognition Letters. 1999. N 20. P. 207–213. doi:10.1016/S0167-8655(98)00115-9
 26. **Jong-hyun Ryu, Sujin Kim, Hong Wan.** Pareto Front Approximation with Adaptive Sum Method in Multi-objective Simulation Optimization // Proc. of the 2009 Winter Simulation Conf. (WSC), Austin, December 13–16, 2009. P. 623–633.

UDC 004.932.2

doi:10.15217/issn1684-8853.2017.3.18

Impulse Noise Filtration based on Community Detection in GraphsBelim S. V.^a, Dr. Sc., Phys.-Math, Professor, sbelim@mail.ruLarionov S. B.^a, Post-Graduate Student, me@stas-larionov.ru^aF. M. Dostoevskiy Omsk State University, 55a, Mira Av., 644077, Omsk, Russian Federation

Introduction: Recognition and removal of impulse noise in graphical images is a pressing problem in signal processing theory and practice. The existing noise reduction algorithms either lack sufficient accuracy or change the noise-free parts of the image. **Purpose:** The goal is to develop a noise filtration algorithm using graphs. **Results:** On the base of the proposed algorithm, a way was found to restore an image distorted by impulse noise. The image is represented as a non-oriented weighted graph. Each pixel of the image corresponds to a vertex in the graph. A community detection algorithm was applied to the graph. The noised pixels are those which correspond to communities containing only one vertex. The method was tested by a computer experiment conducted on grayscale and colored artificial images, as well as on photos. The proposed method is better than a median filter by 20% regardless of the noise level. Its higher efficiency is caused by the fact that most filters change all the image pixels, while the suggested method finds and restores only the noised ones. The dependence of the effectiveness of the proposed method on the image noise percentage is presented.

Keywords — Impulse Noise, Image Filtration, Community Detection in Graphs.

References

1. Pitas I., Venetsanopoulos A. *Nonlinear Digital Filters: Principles and Applications*. Boston, Kluwer Academic Publishers, 1990. 391 p.
2. Huang T. S. *Two-dimensional Digital Signal Processing II: Transforms and Median Filters*. Berlin a.o: Springer, 1981. 391 p.
3. Boyle R., Sonka M., Hlavac V. *Image Processing, Analysis, and Machine Vision, First Edition*. Cambridge, University Press, 2008. 920 p.
4. Mitra S., Sicuranza G. *Nonlinear Image Processing*. Academic Press, 2001. 455 p.
5. Buades A., Morel J. A Non-Local Algorithm for Image Denoising. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition 2*, Washington DC, 2005, pp. 60–65.
6. Dabov K., Foi A., Katkovnik V., Egiazarian K. Image Denoising by Sparse 3-D Transform-domain Collaborative Filtering. *IEEE Transactions on Image Processing*, 2007, no. 8(16), pp. 2080–2095. doi:10.1109/TIP.2007.901238
7. Chan R., Ho C., Nikolova M. Salt-and-pepper Noise Removal by Median-type Noise Detectors and Detail-preserving Regularization. *IEEE Transactions on Image Processing*, 2005, no. 10(14), pp. 1479–1485. doi:10.1109/TIP.2005.852196
8. Kam H. S., Tan W. H. Noise Detection Fuzzy (NDF) Filter for Removing Salt and Pepper Noise. *IVIC 2009: Visual Informatics: Bridging Research and Practice*, Kuala Lumpur, 2009, pp. 479–486. doi:10.1007/978-3-642-05036-7_45
9. Najeer A. J., Rajamani V. Design of Hybrid Filter for Denoising Images Using Fuzzy Network and Edge Detecting. *American Journal of Science Research*, 2009, no. 3, pp. 5–14.
10. Abreu E., Lightstone M., Mitra S. K., Arakawa S. K. A New Efficient Approach for the Removal of Impulse Noise from Highly Corrupted Images. *IEEE Transactions on Image Processing*, 1996, no. 6(5), pp. 1012–1025. doi:10.1109/83.503916
11. Garnett R., Huegerich T., Chui C., He W. A Universal Noise Removal Algorithm with an Impulse Detector. *IEEE Trans Image Process*, 2005, no. 14(11), pp. 1747–1754. doi:10.1109/TIP.2005.8572.61
12. Belim S. V., Seliverstov S. A. Hierarchy Analysis Method as a Way to Detect Impulse Noise on Images. *Informatsionnye tekhnologii* [Information Technologies], 2015, no. 4, pp. 251–258 (In Russian).
13. Belim S. V., Mayorov-Zilbernegel A. O. Algorithm for Searching the Broken Pixels and Eliminating Impulse Noise in Images Using a Method of Association Rules. *Nauka i obrazovanie* [Science & Education], 2014, no. 12. Available at: <http://technomag.bmstu.ru/doc/744983.html> (accessed 08 February 2017) (In Russian). doi:10.7463/1214.0744983
14. Belim S. V., Kutlunin P. E. Impulse Noise Detection in Images Using a Clustering Algorithm. *Vestnik komp'iuternykh i informatsionnykh tekhnologii* [Herald of Computer and Information Technologies], 2016, no. 3, pp. 3–10 (In Russian). doi:10.14489/vkit.2016.03.pp.003-010
15. Larionov I. B. Multidimensional Linear Manifolds as a Way to Restore Graphical Information. *Matematicheskie struktury i modelirovanie*, 2010, no. 1(21), pp. 24–31 (In Russian).
16. Larionov I. B. Kohonen Maps as a Way to Restore the Multimedia Information. *Zhurnal radioelektroniki*, 2010, no. 10, p. 29 (In Russian).
17. Mironovsky L. A., Slaev V. A. *Strip-method for Image and Signal Transformation*. Berlin, De Gruyter, 2011. 160 p.
18. Mironovsky L. A., Slaev V. A. Two-sided Noise-proof Strip-transform and its Root Images. *Izmeritel'naia tekhnika* [Measuring Technology], 2012, no. 10, pp. 6–10 (In Russian).
19. Mironovsky L. A., Slaev V. A. Strip-method of Noise-proof Image Transformation. *Izmeritel'naia tekhnika* [Measuring Technology], 2006, no. 8, pp. 6–12 (In Russian).
20. Belim S. V., Mayorov-Zilbernegel A. O. Image Restoration with Static Gaps on The Basis of Association Rules. *Vestnik komp'iuternykh i informatsionnykh tekhnologii* [Herald of Computer and Information Technologies], 2014, no. 12, pp. 18–23 (In Russian). doi:10.14489/vkit.2014.12.pp.018-023
21. Belim S. V., Seliverstov S. A. The Analytic Hierarchy Method-Based Algorithm for Restoring Broken Pixels on the Noisy Images. *Nauka i obrazovanie* [Science & Education], 2014, no. 11. Available at: <http://technomag.bmstu.ru/doc/742145.html> (accessed 20 January 2017) (In Russian). doi:10.7463/1114.0742145
22. Belim S. V., Larionov S. B. An Algorithm of Image Segmentation Based on Community Detection in Graphs. *Komp'iuternaia optika* [Computer Optics], 2016, no. 6, pp. 904–910 (In Russian). doi:10.18287/2412-6179-2016-40-6-904-910
23. Newman M. E. Analysis of Weighted Networks. *Physical Review E*, 2004, vol. 70, no. 5, p. 056131. doi:10.1103/PhysRevE.70.056131
24. Clauset A., Newman M. E. J., Moore C. Finding Community Structure in Very Large Networks. *Physical Review E*, 2004, vol. 70, no. 6, p. 066111. doi:10.1103/PhysRevE.70.066111
25. DiGesù V., Staravoitov V. V. Distance-based Functions for Image Comparison. *Pattern Recognition Letters*, 1999, vol. 20, pp. 207–213. doi:10.1016/S0167-8655(98)00115-9
26. Jong-hyun Ryu, Sujin Kim, Hong Wan. Pareto Front Approximation with Adaptive Sum Method in Multiobjective Simulation Optimization. *Proc. of the 2009 Winter Simulation Conf. (WSC)*, Austin, 2009, pp. 623–633.